

(43) Date of A Publication **07.06.2000**

(21) Application No **9826640.6**

(22) Date of Filing **03.12.1998**

(71) Applicant(s)
Red Fig Limited
(Incorporated in the United Kingdom)
42 Dover Street, LONDON, W1X 3RB,
United Kingdom

(72) Inventor(s)
Julian Goring Archdale Ellison
Mark Gregory Smith

(74) Agent and/or Address for Service
Venner Shipley & Co
20 Little Britain, LONDON, EC1A 7DH,
United Kingdom

(51) INT CL⁷
H04Q 7/22

(52) UK CL (Edition R)
H4L LDGX L1H10

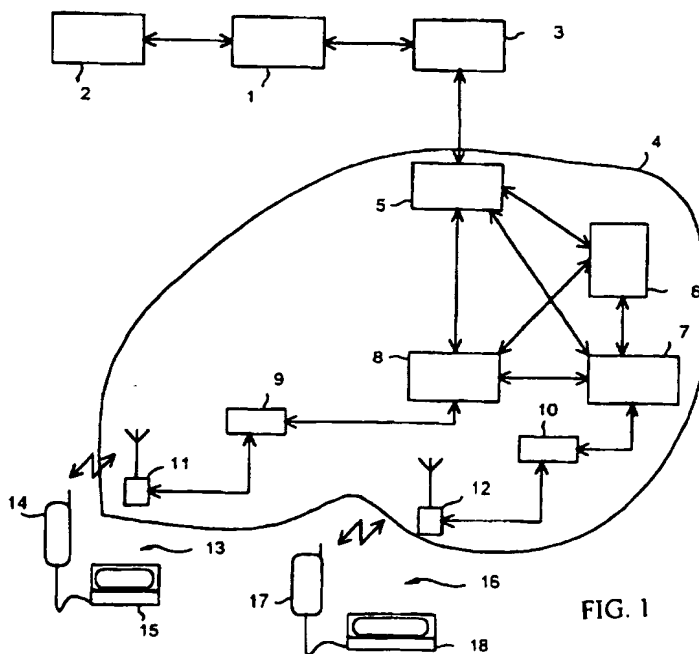
(56) Documents Cited
EP 0859500 A2 **WO 98/11744 A1**

(58) Field of Search
UK CL (Edition Q) **H4L LDGX LDPP**
INT CL⁶ **G06F 17/30 , H04M 3/50 , H04Q 7/22**
ONLINE: **EPODOC, WPI, PAJ**

(54) Abstract Title
Browsing the Internet using a mobile telephone

(57) Variable data for HTML pages is transmitted using the messaging service of a mobile telephone network (4). At clients (13, 16), data in received messages is merged into HTML page templates to create pages from display using a web browser program. The messages are transmitted by a server (1) and cached in the clients (13, 16). When a URL is generated by the web browser program, it is analysed in the client (13, 16) and, if a response is need from the server (1) or information needs be recorded at the server (1) or sent to another location, a message service message bearing URL data is transmitted from the client (13, 16). The server (1) processes these messages and takes the appropriate actions in response.

The system can be used to provide passive information dissemination and interactive services such as flight booking or multi-player games.



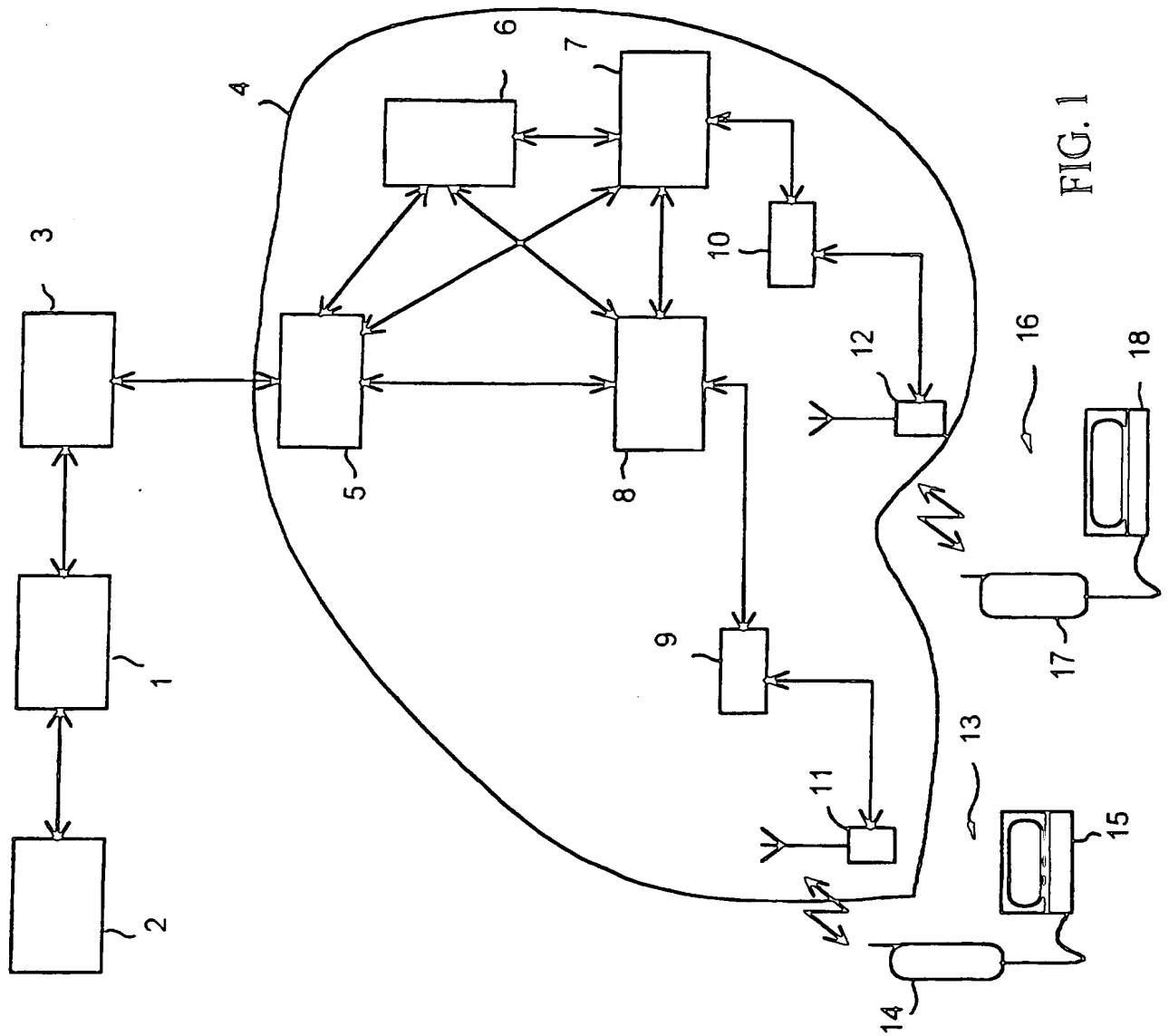
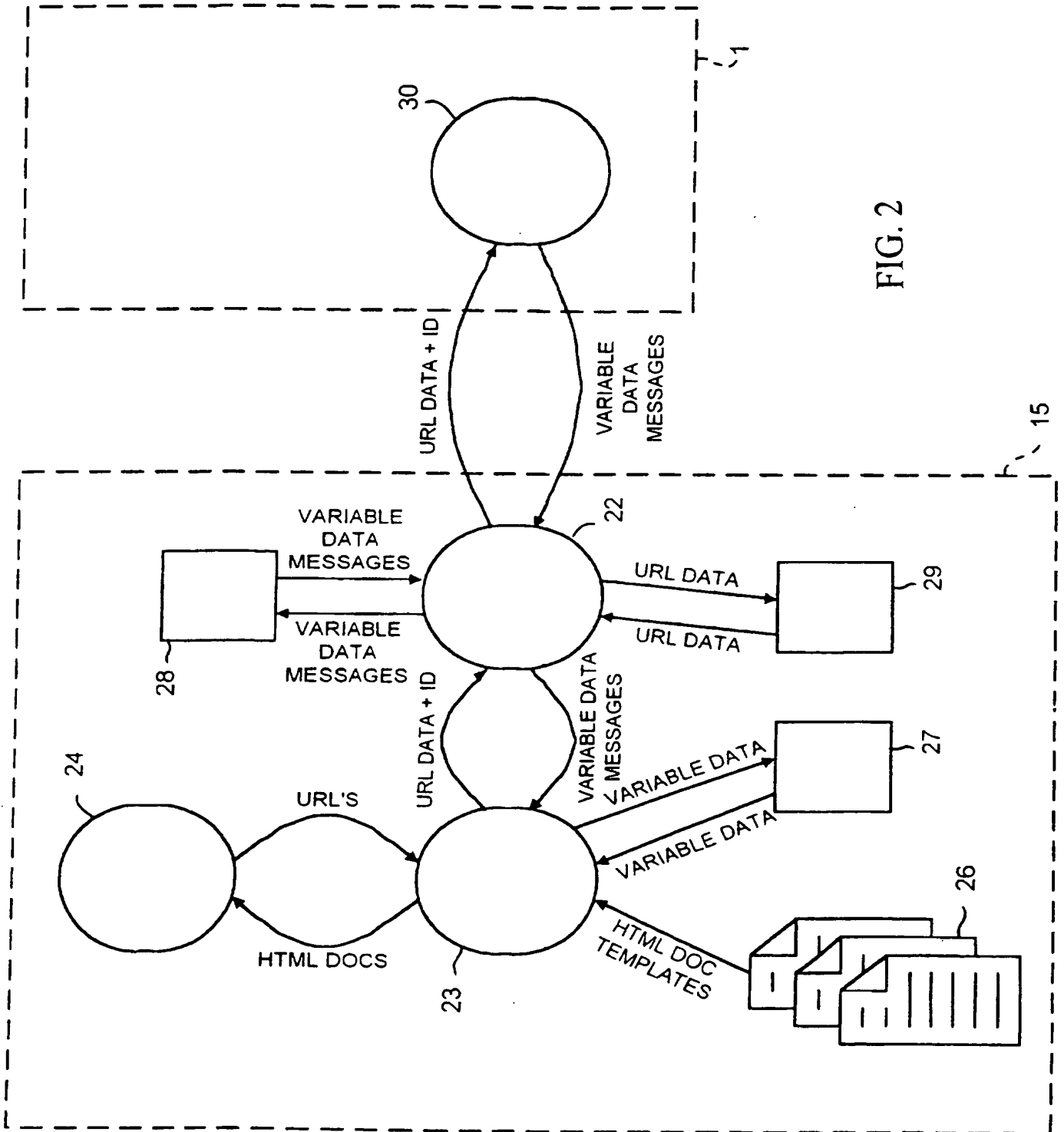


FIG. 1



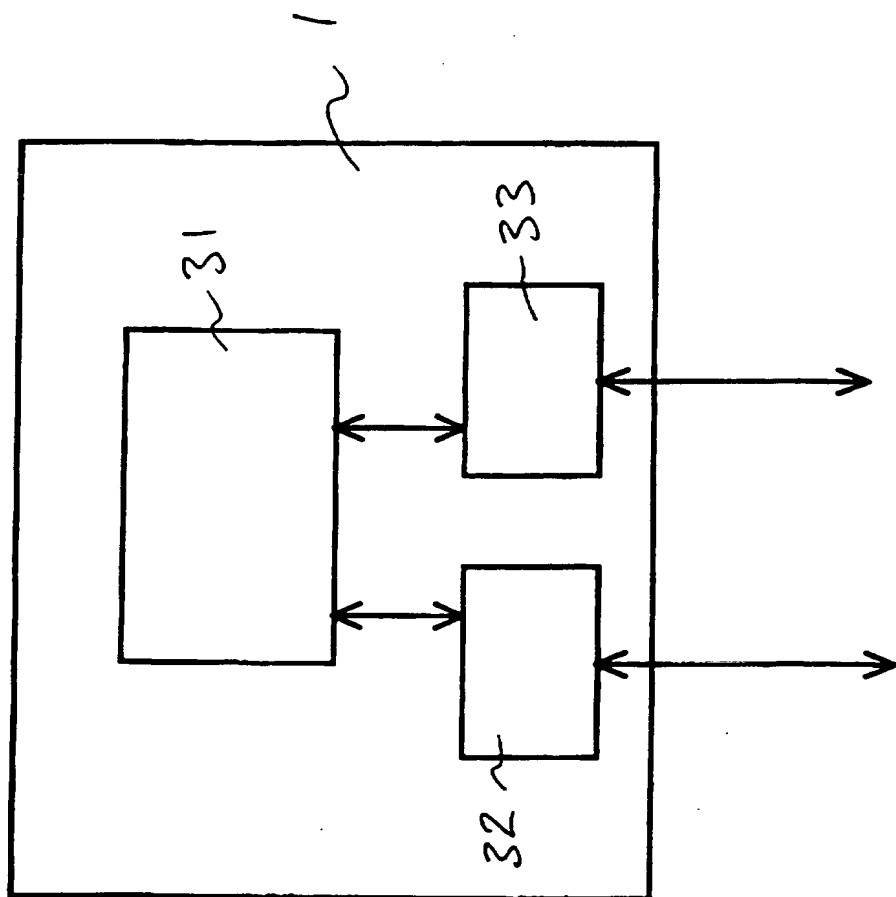


FIG. 3

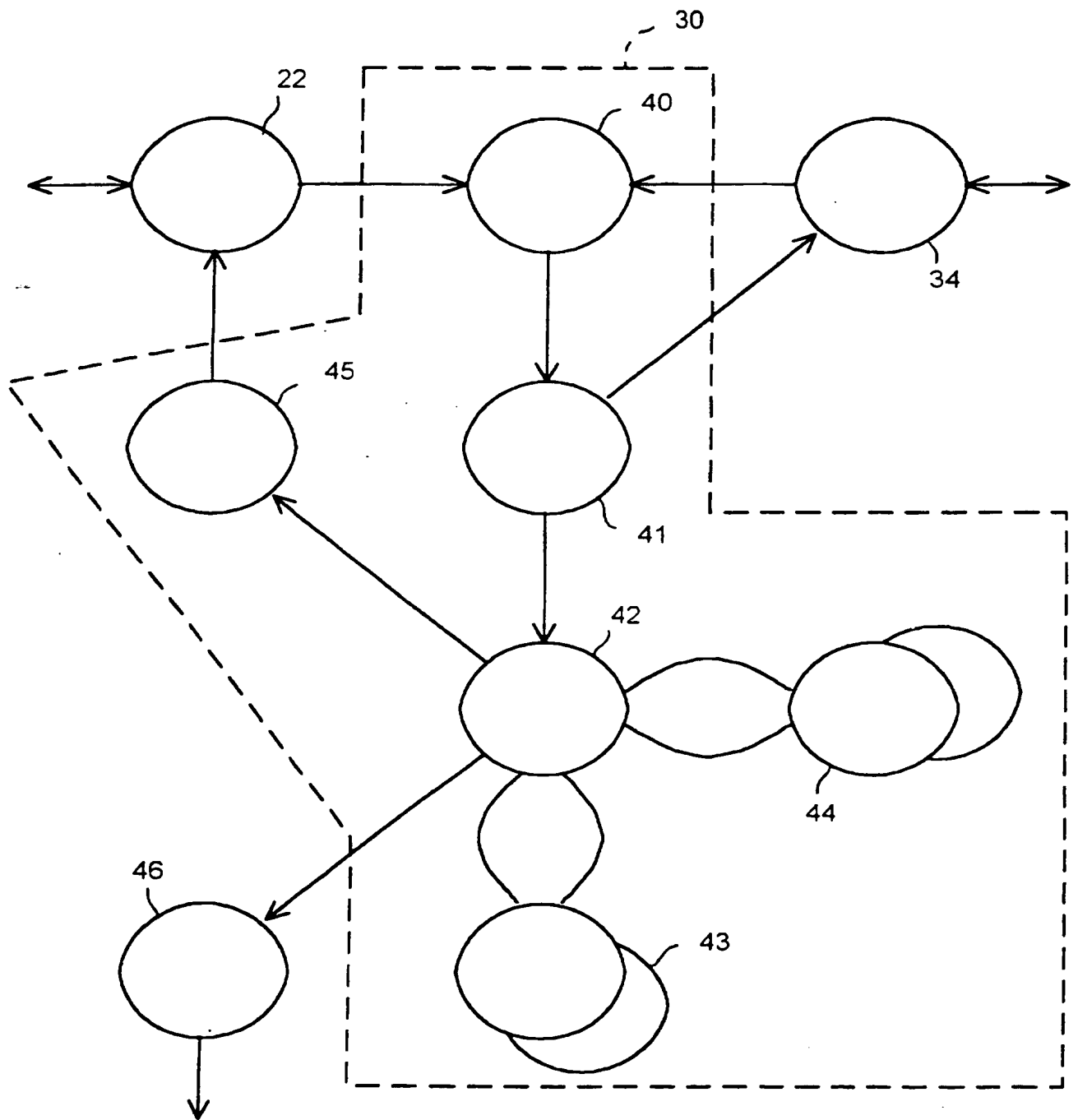


FIG. 4

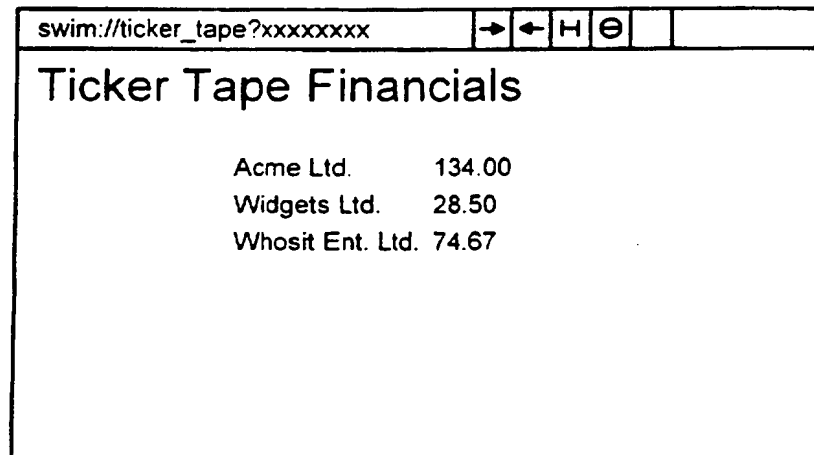


FIG. 5

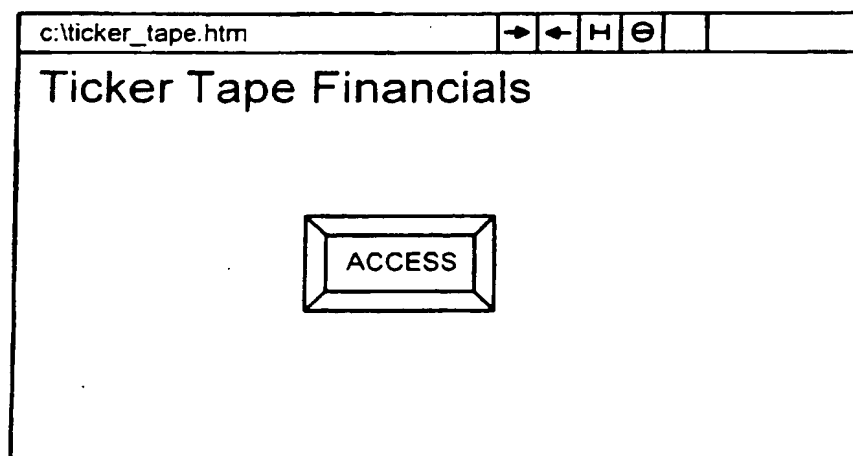


FIG. 6

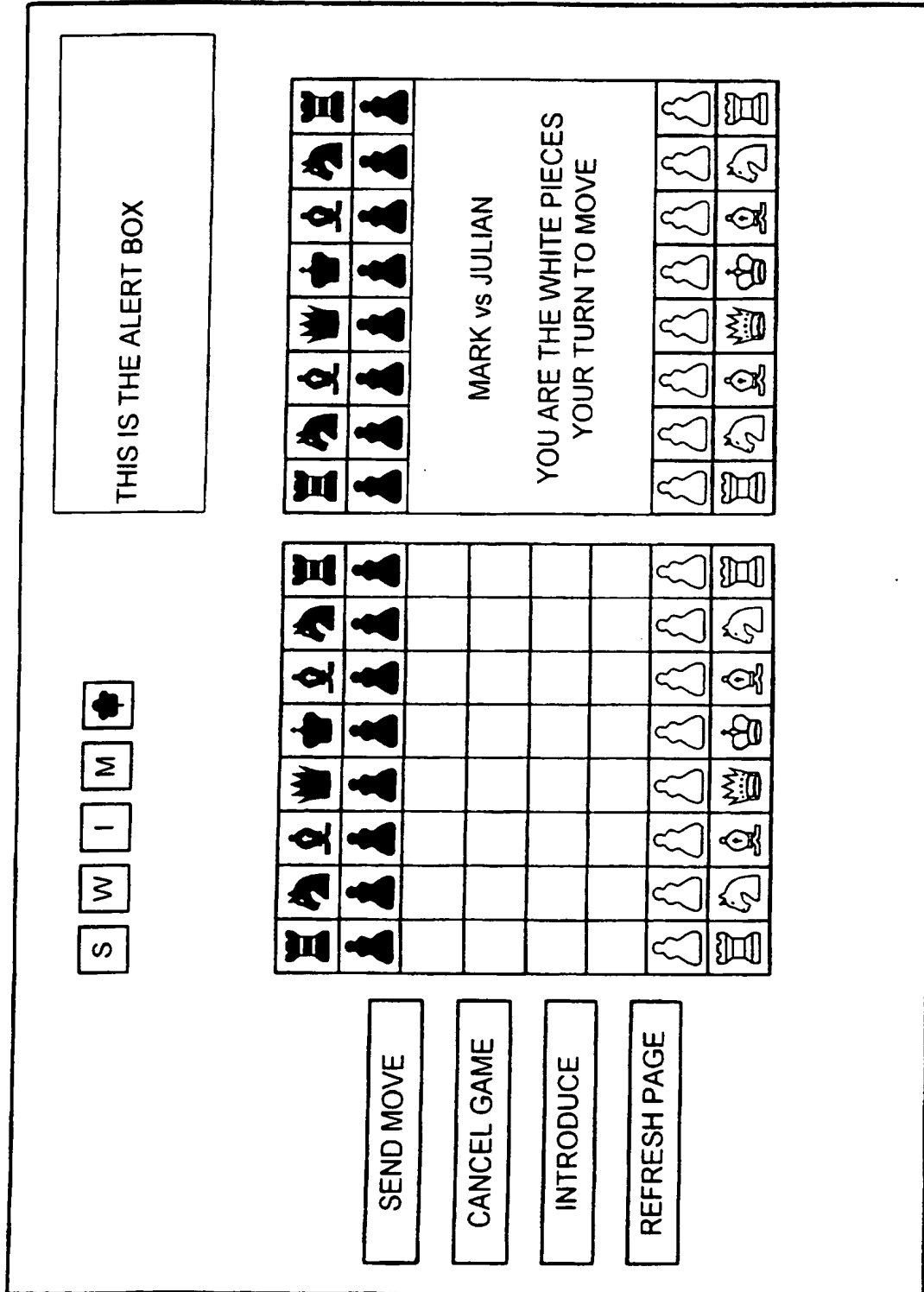


FIG. 7

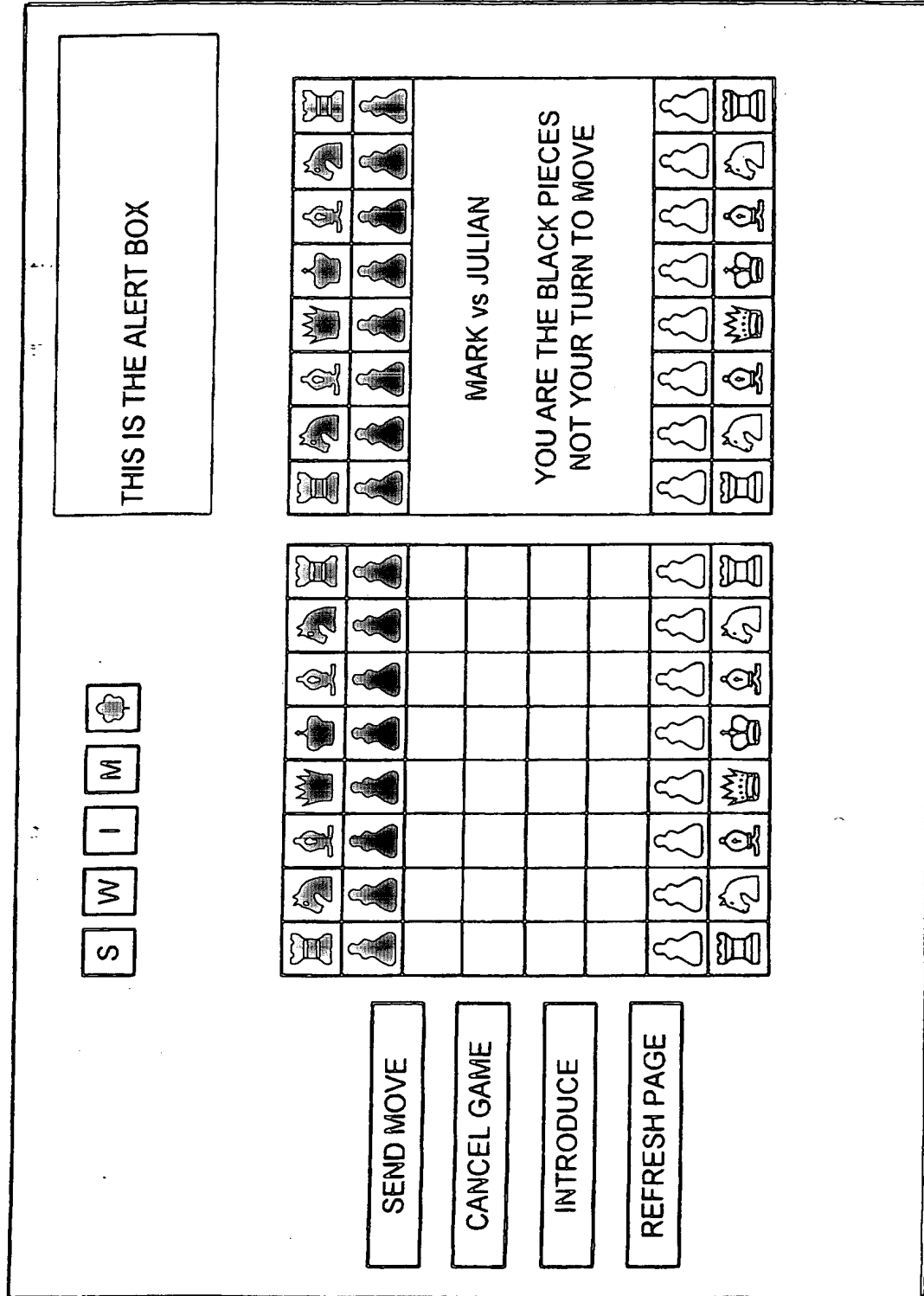


FIG. 8

Data Communication

Description

The present invention relates to data communication.

Recent developments in mobile telephones and portable computers have made the use of browser-based applications in the mobile environment a possibility. It is possible to use a portable computer and a GSM mobile phone to connect to the web sites on the Internet. In this situation, the mobile telephone replaces a pstn dial-up connection.

However, a problem arises in that mobile telephone connections are not as stable as land line connections. For instance, contact can be lost when a mobile telephone user passes into a tunnel. This is not an infrequent occurrence when travelling by train. Furthermore, call costs for mobile telephones tend to be higher than for pstn-only calls.

It is an aim of the present invention to overcome this problem.

The present invention resides in using message service messages from a mobile telephone network to produce web browser pages. This overcomes the problem of interruptions that occur when a mobile telephone system traffic channel is used for receiving web browser data. Whilst the present invention has particular application to the case where a user is mobile, it is not limited thereto and may be embodied using apparatus that is not designed for mobile use together with a mobile telephone.

"Message service" means a data communication service provided in a mobile telephone network which transmits data independently of speech traffic call handling procedures. An example of such a service is the SMS (short message service) provided in GSM networks.

In accordance with the present invention as hereinbefore stated, the applicant seeks protection for a method of communicating information comprising the steps of receiving, by means of mobile telephone means, a message service message from a mobile telephone network, generating a browser page in dependence on the received message, and displaying the generated page using a web browser.

Preferably, the browser page is generated by merging data in said message with a template page. This is advantageous in the case of existing mobile telephone networks because it reduces the amount of data that must be transmitted using the narrow bandwidth message services available.

Preferably, such a method according to the present invention comprises the steps of: caching message service messages from a mobile telephone network and, in response to generation of a URL signal by means of the browser, updating a database of variable data elements associated with a plurality of page templates in dependence on the content of the cached messages.

In accordance with the present invention as hereinbefore stated, the applicant seeks protection for a method of communicating information comprising the steps of generating a URL signal by means of a web browser, generating a message service message in dependence on said URL signal and transmitting said message to a mobile telephone network using mobile telephone means.

Preferably, such a method comprises caching data in a first URL signal and, in response to generation of a second, subsequent URL signal, generating said message service message in dependence the data in both URL signals. In this way, the number of messages that need to be transmitted from a mobile terminal can be kept to a minimum. The first and second URL signals may be separated in time by further URL signals which are also cached and used for generating the message service message.

The foregoing methods can be combined to provide an interactive service, in which a transmitted message relating to a URL elicits a response also relating to the browser page from which the URL originated.

"URL signal" means a signal serving the same function as a URL signal in a world wide web system. Such a signal need not be in the same format as a URL signal used in the world wide web.

In accordance with the present invention as hereinbefore stated, the applicant seeks protection for a method of disseminating information comprising the steps of receiving a first message performing data processing in dependence on the contents of the message to generate a second, message service message and transmitting the second, message service
5 message through a mobile telephone network. The first message may be a message service message. However, other types of messages, e.g. e-mail or URL data received via a web server may be used as the first message.

The second message may be addressed to the sender of the first message. However, it may
10 be addressed to another user thereby providing a user-to-user messaging function. Additionally, outputs other than message service messages may be produced. The messages may be for billing purposes or for controlling other services such as conference call facilities or flight or hotel booking.

15 Preferably, such a method comprises the steps of analysing the first message to determine to which of a set of supported browser pages it relates, passing data in the message to an object instance related to said page and to the sender of the first message, performing a method of the object on the basis of said data so as to generate output data and generating the second message using said output data. Some form of list or collection identifying the
20 browser pages may be stored for access in performing the method. More preferably, such a method includes maintaining a record, for each supported browser page, of the state of the page for each user for whom there is an object instance related to the page.

In accordance with the present invention as hereinbefore stated, the applicant seeks
25 protection for a client apparatus comprising means configured for receiving message service message data from mobile telephone means, a web browser and means for generating a browser page in dependence on received short message service message data.

The client apparatus may include mobile telephone means, for instance a GSM PCMCIA
30 card or be implemented using a communicator, such as the Nokia 9000, which combines data processing and mobile telephone functions.

If a separate mobile telephone is used, the means configured for receiving message service message data from mobile telephone means may comprise a cable or an infrared data link between a portable computer and the mobile telephone means.

5 Preferably, such an apparatus includes storage means storing a template for a browser page and the means for generating a browser pages is configured for modifying the template in dependence on received message service data to produce a browser page.

10 Preferably, such apparatus is configured to provide a first means for caching the contents of received short message service messages and a second means for receiving cached short message service message contents from the first process, merging data comprised in said contents with an page template to produce an browser page and supplying the browser page to the browser. More preferably, the second means supplies URL data from the browser to the first means and the second means generates short message service messages
15 in dependence on URL data from the second means.

In accordance with the present invention as hereinbefore stated, the applicant seeks protection for a server apparatus for a data dissemination system employing a messaging service of a mobile telephone network and a mobile terminal with a web browser, the
20 apparatus comprising receiving means for receiving messages, dispatching means for dispatching message service messages to a mobile station of a mobile telephone network, and processing means for generating a message service message to be dispatched by the dispatching means in dependence a message service message received by the receiving means. The receiving means is preferably configured for receiving message service
25 messages.

Preferably, the processing means comprises a database of users of clients, means for analysing a message, received by the receiving means, to determine to which of a set of supported browser pages it relates, and means for generating object instances, each
30 instance being related to a browser page and a user, wherein said objects have a method for generating output data, to be included in a message service message for dispatch by the dispatching means, on the basis of a received message.

A feature of message services, such as the GSM short message service, is that subscribers can conduct calls during the transmission and reception of short message service messages. Consequently, users of a system according to the present invention can talk to each other whilst viewing the same live data using their browsers. Thus, users playing the above-described chess game can converse while playing the game.

An embodiment of the present invention will now be described, by way of example, with reference to the accompanying drawings, in which:-

- Figure 1 shows a system according to the present invention;
- 10 Figure 2 is a data flow diagram illustrating the operation of a client of Figure 1;
- Figure 3 is a functional block diagram of the server of Figure 1;
- Figure 4 is a data flow diagram illustrating the configuration and operation of the server.
- Figure 5 shows a browser page for a financial ticker tape service;
- Figure 6 shows a ticker tape service access browser page; and
- 15 Figure 7 and 8 shows chess game browser pages for first and second players.

Referring to Figure 1, a system according to the present invention comprises a server 1, a control station 2 connected to the server 1 via the Internet, a short message service centre 3 connected to the server 1 by a leased line, and a GSM mobile telephone network 4. The GSM network 4 includes a short message service gateway 5, a home location register 6 and first and second mobile service switching centres 7, 8. The mobile service switching centres 7, 8 are connected respectively to first and second base station controllers 9, 10 (in practice each mobile service switching centre will usually be connected to a plurality of base station controllers). The base station controllers 9, 10 are connected respectively to first and second base transceiver stations 11, 12 (in practice each base station controller will usually be connected to a plurality of base transceiver stations). A first client 13 comprises a first mobile station 14 connected by a serial link to a first portable computer 15. The first client 13 is located in the cell covered by the first base transceiver station 9. A second client 16 comprises a second mobile station 17 connected by a serial link to a second portable computer 18. The second client is located in the cell covered by the second base transceiver station 12. In this example, the first and second mobile stations 14, 17 have built-in modems such as are found in the Ericsson SH888 and Nokia 8810. Conveniently, the portable computers 15, 18 will be handheld machines such as WindowsCE or Psion

Series 5 machines. There may be a plurality of control stations 2, each dealing with one or more groups of users.

The routing of short message service messages is conventional and details can be obtained
5 from Mouly, M et al., "The GSM system for Mobile Communications", ISBN 2- 9507190-
0-7.

Referring to Figure 2, the first portable computer 15 supports a proxy server process 22, a
protocol handler process 23 and a web browser 24. The proxy server process 22 runs
10 whenever the computer 15 is on. However, the protocol handler process 23 will only be
set running when its functions are required.

The protocol handler process 22 processes URLs submitted from the browser 24 and
returns HTML pages to the browser 24. The HTML pages are created by the protocol
15 handler process 23 by merging locally generated data, e.g. user name, and data, received
from the server 1 in short message service messages, with HTML page templates.

The proxy server process 22 caches short message service messages from the server 1
which relate to the HTML page templates. These messages have a characteristic header in
20 their user data portion. It also performs some local functions in response to submitted
URLs, and generates and sends short message service messages to the server 1 using the
first mobile station 14. The messages sent by the proxy server process 22 are generated on
the basis of submitted URLs passed from the protocol handler process 23.

25 A short message service server process 30 sends short message service messages to the first
client 15 in response to various events which will be described below. These messages
contain variable data for HTML pages for which the user has templates 26. The raw short
message service message contents are stored by the proxy server process 22 in a cache 28.

30 When a user submits a URL using the browser 24, the browser 24 determines whether it
refers to a page according to the present invention from the presence or absence of a
swim:// prefix, instead of, for example, the conventional http:// prefix. If the URL has
the swim:// prefix, the protocol handler process 23 is started and the URL is passed to it.

"swim://" URLs comprise a page identifier and one or more name-value pairs. The name-value pairs may include data to be communicated to the server process 30 or stored locally, and directives for defining the handling of the URL by the protocol handler process 23 and the proxy server process 22.

If the URL includes a directive indicating that no variable data or only locally generated variable data is required, the protocol handler process 23 simply merges the identified HTML page template 26 with any identified locally generated variable data from a local data store 27 and returns the resulting HTML page to the browser 24.

If, however, the URL indicates that data from the server 1 is required, the protocol handler process 23 passes the URL (or at least that part of it as is required by the proxy server process 22) to the proxy server process 22 together with a message ID. If the URL does not require an *ad hoc* response from the server process 30, the proxy server process 22 transfers the contents of its message cache 28 to the protocol handler process 23.

In this case, the protocol handler process 23 analyses any messages received from the proxy server process 22 and updates the variable data in its local data store 27. The protocol handler process 23 then retrieves the variable data for the submitted URL from its local data store 27 and merges it with the relevant HTML page template. The resultant HTML page is then sent to the browser 24.

A proxy server directive is provided that causes data in a URL to be stored in the message cache 28 as a pseudo short message service message. This data will then be sent back to the protocol handler process 23 as if it had been received from the server process 30. In this way, the contents of the protocol handler process's data store 27 can be amended to include data input by the user and to update controls on the displayed page without the need to contact the server 1 and receive a reply therefrom.

If the URL data received by the proxy server process 22 indicates that the server 1 must be contacted, the proxy server process 22 builds a short message service message using the

data elements of the URL and the message ID and then sends the message to the server process 30 using the mobile station 13 and the short message service of the network 5.

5 The structure and operation of the server 1 and the server process 30 will be described in detail below. However, if a response is not required from the server 1, the proxy server process 22 immediately sends the contents of its message cache 28 to the protocol handler process 23 which proceeds as described above.

On the other hand, if a response is required from the server process 30, the proxy server
10 process 22 monitors incoming short message service messages until it receives a complete message bearing the ID of the message that it earlier sent to the server process 30. When this reply message has been received, the proxy server process 22 passes the contents of its message cache 28, which now include the reply message, to the protocol handler process 23 which processes the message cache data as described above and generates the new HTML
15 page using the data returned by the server process 30.

In order to minimise short message service traffic, URL data can also be cached by the proxy server process 22 in a URL data cache 29 until a URL containing an explicit submit directive is received from the protocol handler process 23. This mechanism would typically
20 be employed where a user needs to click several times on an HTML page to make a selection and then click on a submit button. In this case, the initial clicks would cause a sequence of URLs to be sent to the proxy server process 22 which each contain a cache directive and data. This data would then be stored in the URL data cache 29. Clicking on the submit button would cause a URL with a submit directive to be sent to the proxy
25 server process 22 which would then build an short message service message using the cached data and send it to the server 1. A variant of this directive contains a count value and triggers a submit to server 1 when the number of received URL's, containing the directive and originating from the same HTML page, reaches the count value.

30 Since conditions at the server 1 may change with time and the server process 30 can send variable data to clients 13, 16 without a specific request being made, it is desirable that the browser 24 resubmits the current URL at intervals so that the displayed page remains up to date.

The second client 16 is similarly configured and interacts with the server in the same manner.

5 The foregoing gives an overview of the functionality of an example of a client according to the present invention. Generally, the directives need to be defined when the protocol handler process 23 and the proxy server process 22 are being designed. However, the data elements will be defined when the HTML page is being designed.

10 Referring to Figure 3, the server 1 comprises a processor 31 including a CPU, ROM, RAM and disk drives, an Internet interface 32 and an short message service interface 33. The server 1 is implemented using a Sun Sparc machine.

Referring to Figure 4, a web server 34 and the short message service server process 30 are
15 implemented as processes on server 1. The web server 34 is provided so that the short message service server 35 can be controlled from a web browser over an Internet connection. The short message service server 30 is implemented using an object-oriented database system.

20 Each user of the system is represented by an instance of a user class and includes, *inter alia*, the user's mobile telephone number. Users are members of one or more groups represented by instances of a group class which hold a list of the HTML pages available to the members of the group. A group may contain only one user.

25 The groups may be employees of the same company, a group of friends, or some other collection of people with an interest in common. A user may permit his details, stored in the database, to be available to other users. This makes it possible for users to create new groups by inviting other members to join. For instance, a user with an interest in surfing might want to set up a service providing information about surf conditions at different
30 beaches. To do this he would design the necessary HTML pages and control objects. The control objects would be uploaded to the server 1. The user could then send e-mails to other users inviting them to join the new group and then e-mail the HTML pages to those who responded positively. Of course, if a service is already established, it would only be

necessary to send the HTML pages to new members of the group for the service. The mode of distribution of the HTML pages is not important, e-mail being given as an example only. Another method of distribution would be to include suitable HTML pages in the free disk on the cover of a computing magazine. These pages would include the HTML pages for the service and a special HTML page for registering for the service with the server 1.

Each page that can be displayed by a client 13, 16 of the system has associated with it a class. The class defines the properties of the page needed by the server 1 and the logic associated with pages of that type in the form of methods. For each page class, there is a respective class handler object instance.

When a short message service message is received from the proxy server process 22 of a client 13, 16, it is passed to a routing object instance 40. The routing object instance 40 identifies the type of page to which the message relates and routes it to the class handler object instance 41 for that type of page. The message contains the identity of the user sending the message and, using this information, the class handler object instance 41 searches through a linked list of instances of objects 42 of the class representing the page for an instance including the user as its or one of its owners.

20

If an page object instance 42 belonging to the user is not found, the class handler object 41 generally instantiates a new page object instance 42. However, messages not requiring the instantiation of the new page object instance 42 are permitted. If an new instance is instantiated, the page object's constructor will instantiate one or more page representation object instances 43. The page representation object instances 43 store the state of the page as presented to the or each user. In some cases, the same page representation object instance 43 will apply to all of the owners and, in other cases, each user will be seeing a different display and, accordingly, respective page representation object instances 43 are required. Page object instances 42 may also have access to the user objects 44 for their owners so that their operation can be tailored according to their owners' preferences.

30

Once the page object instance 42 has been instantiated, or has been found in the search by the page handler object instance 41, it is passed the message which it then processes to

produce an output. The output will usually be a variable data to be sent back to the client 13, 16 that initiated the processing or to be sent to another client 13, 16 whose user belongs to the same group. In these cases, the variable is passed to a short message service message output process 45 which constructs short message service messages to convey the variable data to its destination. However, the output could be to many other destinations 46, e.g. a telephony switch for providing conference call facilities to the owners of the page class object instance 42, or the web server 34 for reporting to a group supervisor.

The short message service server 30 does not only respond to short message service messages from clients 13, 16. An update of variable data for a page may be received from the control station 2 via the web server 34. Such a message takes the form of the identity of the page, the group to which the update relates and the update itself. The message is passed to the routing object instance 40. The routing object instance 40 identifies the type of page to which the message relates and routes it to the class handler object instance 41 for that type of page. The message contains the identity of the page's user through the group identification and, using this information, the class handler object instance 40 searches through a linked list of instances of page objects 42 for an instance including the users of that group as its owners.

If a page object instance 42 belonging to the group is not found, the class handler object 41 instantiates a new page object instance 42. The page object's constructor will instantiate one or more page representation object instances 43 as necessary.

Once the page object instance 42 has been instantiated, or has been found in the search by the page handler object instance 41, it is passed the update message. The page object instance 42 then outputs the updated variable data in the appropriate name-value pairs to the short message service message output process 45 with instructions to send it to each member of the group. The short message service message output process 45 then sends the necessary messages to the group members so that the updated variable data is available next time a user selects the URL for the page or the page is refreshed.

The page object instances 42 will be destroyed when their persistence is no longer required.

A registration HTML page may be made available, this would be treated at the server 1 as described above but the new user would be treated by default as a member of a global group of users. The page object instance 42 would create a new user record in the database in response to the new user supplying personal information, e.g. name, mobile telephone
5 number, e-mail address and credit card details, using the registration HTML page.

Examples of applications of the present invention will now be described.

The first example is a financial ticker tape service.

10 A user will have a template 26 for the HTML page shown in Figure 5 stored in his client 13, 16. It can be seen that the page has a plurality of rows, each having the name of a company and the price of one ordinary share. All of the information on the page is unchanging save for the prices themselves. The template 26 will therefore have all of the
15 text, except the prices, and formatting codes. The positions of the prices are held by respective place holders in the template 26.

The provider of the service has access to the server 1 via the web server 34 and can use this to add new users to the ticker tape service subscribers group. The ticker tape data is
20 generated automatically by the service provider and supplied at regular intervals to the server 1 via the web server 34.

A process at the web server 34 passes this data to the router object 40 which routes it to the page handler object instance 41 for the ticker tape service page. In this case, the page
25 handler object instance 41 will instantiate an instance 42 of the ticker tape page object and then pass the data from the service provider. The ticker tape page object instance 42 outputs the data as name-value pairs to the short message service output process 45 together with the mobile telephone numbers of the subscribers from the user objects 44. The short message service output process 45 then constructs and transmits short message
30 service messages bearing the prices to the client 13, 16. It should be noted that a page representation object instance 43 is not required in this case.

Once the data has been sent to the short message service output process 45, the ticker tape page class object instance is destroyed.

5 The clients 13, 16 terminals of the subscribers to the ticker tape service will therefore receive short message service messages containing the prices. At each client 13, 16, these messages are cached by the proxy server process 22.

10 The URL for the ticker tape service page is rather complex for a user to conveniently type in. Therefore, a ticker tape service access page is provided (see Figure 6) . This page is a conventional HTML page stored locally. As can be seen, the page includes a button for accessing the ticker tape service. When the user clicks on this button, the swim:// URL for the ticker tape service page is submitted. This URL is processed by the protocol handler process 23 and the proxy server process 22. The proxy server process 22 is caused to send the cached messages to the protocol handler process 23 which stores the latest
15 version of the price data in its data store 28. The latest price data is read from the data store 28 by the protocol handler process 23 and merged into the HTML template for the ticker tape service page. The resultant document is then sent to the browser 24 and displayed.

20 If the user does not close the browser 24 or submit another URL, the browser 24 will resubmit the URL for the ticker tape page after a predetermined period has elapsed so that the page will be refreshed with the latest prices.

25 The second example is the processing of a move in a chess game played between the users of the first and second clients 13, 16.

30 The first client 13 is displaying to the page shown in Figure 7 and the second client is displaying the page shown in Figure 8. It can be seen that there are differences between the displayed pages. First and second users are using respectively the first and second clients 13, 16.

At the server 1, an instance of a chess game page object 42 has been instantiated. This object instance 42 embodies the rules of chess and logic for controlling the pages displayed

to the users. Since these pages are different, the chess game object 42 maintains two page representations 43, one for each player. Both page representations 43 record the positions of the pieces.

5 When the first user wishes to move a piece, he first clicks on the bitmap representing the square holding the piece to be moved. The browser 24 responds by generating a URL which it submits to the protocol handler process 23. The URL identifies the page, a location for a "highlighted" bitmap to replace the one clicked on and the identity of the chessboard square represented by the clicked-on bitmap. The URL also contains a
10 directive that causes the location for a "highlighted" bitmap to be stored as a pseudo message in the proxy server's message cache 28 and a directive that causes the chessboard square identity to be stored in the proxy server's URL cache 29. This second directive has associated with it a count value, in this case two.

15 After receiving the URL data, the proxy server process 22 sends the contents of its message cache 28 to the protocol handler process 23. The protocol handler process 23 will then merge the chess game page template with the updated data, which includes the new bitmap location for the clicked-on square, and sends the new page to the browser 24. Consequently, the user is provided with a visual indication of the square on which he
20 clicked.

The first user now clicks on the square to which he wants to move the selected piece. A second URL, which is the same as the first except for the identified square, is generated and sent to the proxy server process 22 via the protocol handler process 23. The bitmap
25 location data is treated as in the first case. However, the caching directive is the same with the same count value, i.e. two. The proxy server process 22 detects that two such directives have now been received and builds an short message service message including the identities of the squares clicked-on by the user and a message ID from the protocol handler process 23. This message is then sent to the server 1.

30

At the server 1, the message is routed to the chess page object instance 42 for the game. This object instance 42 checks that the proposed move is legal by referring to one of the page representations 43. If the move is illegal, the object instance 42 sends a message back

to the first user which causes a message page to be displayed with the text "Error: illegal move" and a link that when selected will cause the chess page to be displayed again.

If the move is legal, the object instance 42 checks for a checkmate condition. The object
5 instance 42 then updates the page representations with the new piece position, swaps "your move" and "opponent's move" text and, if necessary, adds "checkmate - game over". Messages for the users, reflecting the changes to their respective page representations, are then constructed and sent to the users.

10 When the first user receives his message, the proxy server process 22 of the first client 13 caches it until the next refresh is requested by its browser 24, at which point it transfers the contents of its message cache 28 to the protocol handler process 23. The protocol handler process 23 then creates an updated HTML page and sends it to the browser 24 so that the user sees the chess pieces in their new positions, the legend "opponent's move" and, if
15 appropriate, "checkmate - game over".

The message received by the second user is similarly processed by the second client 16. However, the page displayed to the second user will bear the legend "your move", assuming that the last move by the first player did not end the game.

20

In the foregoing description, the server 1 is connected to the short message service centre 3 by a leased line. However, the server 1 may use a mobile station to send messages for clients via the short message service.

25 In a passive embodiment, a client according to the present invention is configured to merge cell broadcast short message service messages in to HTML pages so that these messages can be conveniently viewed using a browser.

Claims

1. A method of communicating information comprising the steps of:
receiving, by means of mobile telephone means, a message service message from a
5 mobile telephone network;
generating a browser page in dependence on the received message; and
displaying the generated page using a web browser.
2. A method according to claim 1, the browser page is generated by merging data in
10 said message with a page template document.
3. A method according to claim 2, comprising the steps of:
caching message service messages from a mobile telephone network; and
in response to generation of a URL signal by means of the browser, updating a
15 database of variable data elements associated with a plurality of page templates in
dependence on the content of the cached messages.
4. A method of communicating information comprising the steps of:
generating a URL signal by means of a web browser;
20 generating a message service message in dependence on said URL signal; and
transmitting said message to a mobile telephone network using mobile telephone means.
5. A method according to claim 4, comprising caching data in a first URL signal and,
in response to generation of a second, subsequent URL signal, generating said message
25 service message in dependence the data in both URL signals.
6. A method of obtaining information comprising transmitting information by a
method according to claim 4 or 5 and receiving information by a method according to
claim 1, 2 or 3, wherein the information in a received message is a response to the
30 information in a transmitted message.
7. A method of disseminating information comprising the steps of:-
receiving a first message;

performing data processing in dependence on the contents of the first message to generate a second, message service message; and

transmitting the second message service message through a mobile telephone network.

5

8. A method according to claim 7, wherein the first message is a message service message from the mobile telephone network.

9. A method according to claim 8, wherein the second message is addressed to the sender of the first message.

10

10. A method according to claim 8 or 9, wherein the first message relates to a browser page displayed at a mobile terminal.

11. A method according to any one of claims 7 to 10, comprising the steps of:-
analysing the first message to determine to which of a set of browser pages it relates;

15

passing data in the message to an object instance related to said page and to the sender of the first message;

performing a method of the object on the basis of said data so as to generate output data; and

20

generating the second message using said output data.

12. A method according to claim 11, including maintaining a record for each supported browser page of the state of the page for each user for whom there is an object instance related to the page.

25

13. A client terminal apparatus comprising means configured for receiving message service message data from mobile telephone means, a web browser and means for generating a browser page in dependence on received short message service message data.

30

14. An apparatus according to claim 13, including mobile telephone means.

15. An apparatus according to claim 14, wherein the means configured for receiving message service message data from mobile telephone means comprises a cable or infrared data link between a portable computer and the mobile telephone means.
- 5 16. An apparatus according to claim 13, 14 or 15, including storage means storing a template for a browser page and the means for generating a browser page is configured for modifying the template in dependence on received short message service data to produce a browser page.
- 10 17. An apparatus according to any one of claims 13 to 16, configured to provide a first means for caching the contents of received short message service messages and a second means for receiving cached short message service message contents from the first process, merging data comprised in said contents with an page template to produce a browser page and supplying the browser page to the browser.
- 15 18. An apparatus according to claim 17, wherein the second means supplies URL data received from the browser to the first means and the second means generates short message service messages in dependence on URL data from the second means.
- 20 19. A server apparatus for a data dissemination system employing a messaging service of a mobile telephone network and a client with a web browser, the apparatus comprising receiving means for receiving messages from a mobile telephone network; dispatching means for dispatching message service messages to a mobile station of a mobile telephone network; and
- 25 processing means for generating a message service message to be dispatched by the dispatching means in dependence a message service message received by the receiving means.
20. A server according to claim 19, wherein the receiving means is configured for
- 30 receiving message service messages.
21. An apparatus according to claim 19 or 20, wherein the processing means comprises:

a database of users of mobile terminals;
means for analysing a message, received by the receiving means, to determine to which of a set of supported browser pages it relates; and

means for generating object instances, each instance being related to a browser
5 page and a mobile terminal user,

wherein said objects have a method for generating output data, to be included in a message service message for dispatch by the dispatching means, on the basis of a received message.

10 22. A client-server data dissemination system including a client apparatus according to any one of claims 13 to 18 and a server according to claim 19, 20 or 21.

23. A client apparatus substantially as hereinbefore described with reference to the accompanying drawings.

15

24. A server substantially as hereinbefore described with reference to the accompanying drawings.

25. A client-server data dissemination system as hereinbefore described with reference
20 to the accompanying drawings.



Application No: GB 9826640.6
Claims searched: all

Examiner: Nigel Hall
Date of search: 21 April 1999

Patents Act 1977
Search Report under Section 17

Databases searched:

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK CI (Ed.Q): H4L (LDPP, LDGX)

Int CI (Ed.6): H04Q 7/22; H04M 3/50; G06F17/30

Other: Online: EPODOC, WPI, PAJ

Documents considered to be relevant:

Category	Identity of document and relevant passage	Relevant to claims
X	EP0859500 A2 (LUCENT) See whole document	7-9
X	WO98/11744 A1 (NOKIA) See whole document	7-9

X	Document indicating lack of novelty or inventive step	A	Document indicating technological background and/or state of the art.
Y	Document indicating lack of inventive step if combined with one or more other documents of same category.	P	Document published on or after the declared priority date but before the filing date of this invention.
&	Member of the same patent family	E	Patent document published on or after, but with priority date earlier than, the filing date of this application.